UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/703,316	11/01/2000	Matthew R. Arnold	13735(YOR92000358US1	1369

7590 04/02/2004

Richard L Catania Esq
Scully Scott Murphy and Presser
400 Garden City Plaza
Garden City, NY 11530

EXAMINER	
TANG, KUO LIANG J	

ART UNIT	PAPER NUMBER
2122	5

DATE MAILED: 04/02/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Offic Action Summary	Application No. 09/703,316	Applicant(s) ARNOLD ET AL.
	Examiner Kuo-Liang J Tang	Art Unit 2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on *01/28/2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-54* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-54* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is in response to the amendment filed on 01/28/2004.

Claims 1-54 are pending, effective date is 11/01/2000.

Claims 1-2 and 19-20 remain rejected under 35 U.S.C. § 102(e).

Claims 3-8, 11, 18, 21-27, 30, 36-45, 48 and 54 remain rejected under 35 U.S.C. § 103(a).

Claims 9, 28 and 46 remain rejected under 35 U.S.C. § 103(a).

Claims 10, 29 and 47 remain rejected under 35 U.S.C. § 103(a).

Claims 12, 31 and 49 remain rejected under 35 U.S.C. § 103(a).

Claim 13 remains rejected under 35 U.S.C. § 103(a).

Claims 14-17, 32-35 and 50-53 remain rejected under 35 U.S.C. § 103(a).

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims 1-2 and 19-20 are rejected under 35 U.S.C. 102(e) as being anticipated by Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy.

As Per Claim 1, Krishnaswamy disclosed:

A sampling-based system for adaptively optimizing a computer program executing in an execution environment, said execution environment including one or more compiler devices for providing various levels of program optimization, said system comprising:

-a runtime measurements sub-system for monitoring execution of said computer program to be optimized, said monitoring including obtaining raw profile data samples and characterizing said raw profile data; (see Column 6, Lines 4-8, "During execution, the dynamic optimization helper 230 **analyzes 330 the executing instructions** and **collects profile data**. As part of this analysis, the dynamic optimization helper 230 searches **for optimization** opportunities.").

-a controller device for receiving said characterized raw profile data from said runtime measurements sub-system and analyzing said data for determining whether a level of program optimization for said executing program is to be performed by a compiler device, said controller generating a compilation plan in accordance with a determined level of optimization; (see Column 5, Lines 43-51, "The **dynamic optimization helper 260** in kernel memory space operates similarly to the **dynamic optimization helper 230** in shared user memory space. It collects profile data regarding the executing computer operating system kernel 220, processes that profile data, and **provides it to the code-rewriting kernel module 240**. The kernel module 240 analyzes the profile data and **generates optimized translations** of portions of the computer operating system kernel 220 from the profile data."). In other word, since the optimization is performed dynamically using the profile data, various levels of optimization is inherently performed depending on the result of the analysis of the profile data. ***and,***

-a recompilation sub-system for receiving a compilation plan from said controller and invoking a compiler device for performing said level of program optimization of said executing program in accordance with said compilation plan. (see Column 5, Lines 51-58, "The **code-rewriting kernel module 240** then writes the optimized translations into a code cache 270 in the kernel memory space. The kernel module 240 also inserts jump instructions into the computer operating system kernel 220 to switch execution flow of control to the optimized translations whenever an optimized instruction within the computer operating system kernel 220 is called.").

As Per Claim 2, the rejection of claims 1 is incorporated and further Krishnaswamy disclosed:

-one or more organizer devices for processing said raw profile data and characterizing said data as meeting a hotness threshold of activity. (see Column 2, Lines 7-11, "The dynamic optimization program analyzes the profile data, looking, for example, for "hot" instruction paths (series of consecutive instructions that are executed often during execution). The dynamic optimization program then optimizes portions of the computer program based on the profile data.") and (see Column 7, Lines 55-57, "If a "hot trace" is identified as beginning at 0x110, the present invention replaces the bundle at 0x110 with a branch jumping to the optimized trace in the code cache 250:"). For a path to be defined as "hot", it must have been executed several times and analyzed to meet or exceed a certain value (threshold) among the profile data.

As Per Claim 19, Krishnaswamy disclosed:

a) sampling said executing computer program to obtain raw profile data samples; (see Column 6, Lines 4-8, "During execution, the dynamic optimization helper 230 **analyzes 330 the**

executing instructions and **collects profile data**. As part of this analysis, the dynamic optimization helper 230 searches **for optimization** opportunities.").

b) characterizing said raw profile data as meeting a threshold criteria; (see Column 2, Lines 7-11, "The dynamic optimization program analyzes the profile data, looking, for example, for "hot" instruction paths (series of consecutive instructions that are executed often during execution). The dynamic optimization program then optimizes portions of the computer program based on the profile data.") and (see Column 7, Lines 55-57, "If a "hot trace" is identified as beginning at 0x110, the present invention replaces the bundle at 0x110 with a branch jumping to the optimized trace in the code cache 250:"). For a path to be defined as "hot", it must have been executed several times and analyzed to meet or exceed a certain value among the profile data. Therefore, threshold is inherent.

c) analyzing said characterized raw profile data for determining whether a level of program optimization for said executing program is to be performed by a compiler device, and generating a compilation plan in accordance with a determined level of optimization; (see Column 5, Lines 43-51, "The **dynamic optimization helper 260** in kernel memory space operates similarly to the **dynamic optimization helper 230** in shared user memory space. It collects profile data regarding the executing computer operating system kernel 220, processes that profile data, and **provides it to the code-rewriting kernel module 240**. The kernel module 240 analyzes the profile data and **generates optimized translations** of portions of the computer operating system kernel 220 from the profile data."). In other word, since the optimization is performed dynamically using the profile data, various levels of optimization is inherently performed. They depend on the result of the analysis of the profile data. ***and,***

d) when optimization is to be performed, invoking a compiler device for optimizing said executing program in accordance with said compilation plan. (see Column 5, Lines 51-58, "The **code-rewriting kernel module 240** then writes the optimized translations into a code cache 270 in the kernel memory space. The kernel module 240 also inserts jump instructions into the computer operating system kernel 220 to switch execution flow of control to the optimized translations whenever an optimized instruction within the computer operating system kernel 220 is called.").

In regard to ***Claim 20***, the rejection of claims 19 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 2.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 3-8, 11, 18, 21-27, 30, 36-45, 48 and 54 are rejected under 35 U.S.C. 103(a) as being unpatentable over Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy in view of Holzle al. (US Patent No. 5,995,754) hereafter Holzle.

As Per Claim 3, the rejection of claim 2 is incorporated and further Krishnaswamy didn't disclose counting a number of samples.

However, Holzle teaches ***a mechanism for counting a number of samples that are taken from the executing program*** as claimed (see Column 2, Lines 52-57, "In the Self system, the determination of whether to re-compile previously compiled code is made based on how many times a specific portion of compiled code, such as a method, has been called. If the method has been invoked more times than a fixed limiting value, then the method is re-compiled."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to count samples. The modification would have been obvious because one of ordinary skill in the art would have been motivated to use the number of samples to help to determine whether to perform the optimization or not.

Krishnaswamy further didn't disclose comparing the number of samples to a predetermined threshold. However, Holzle teaches ***comparing the number of samples to a predetermined sampling size threshold, and in response to the number of samples exceeding said sampling threshold. invoking said organizer device to process said raw profile data.*** (see Column 2, Lines 56-60, "If the method has been invoked **more times than a fixed limiting value**, then the method is re-compiled. The fixed limiting value is essentially a fixed threshold, which reflects the number of times the method is to be invoked before the method is re-compiled to increase efficiency in execution."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to compare a threshold value. The modification would have been obvious because one of ordinary skill in the art would have been motivated to improve the

efficiency of the system by recompiling only method that is being called more times than a fixed limited values.

As Per Claim 4, the rejection of claim 3 is incorporated and further Krishnaswamy didn't disclose raw profile data samples relate to one or more method activations. However, Holzle teaches ***raw profile data samples relate to one or more method activations in said executing program.*** (see Column 2, Lines 52-57, "In the Self system, the determination of whether to re-compile previously compiled code is made based on how many times **a specific portion of compiled code, such as a method, has been called.** If the **method** has been **invoked** more times than a fixed limiting value, then the method is re-compiled."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to have samples relate to method activations. The modification would have been obvious because one of ordinary skill in the art would have been motivated to perform the system optimization by recompiling frequently invoked method.

As Per Claim 5, the rejection of claim 4 is incorporated and further

Krishnaswamy didn't disclose comparing said raw profile data of method activations against a corresponding threshold. However, Holzle teaches ***mechanism for comparing said raw profile data of method activations against a corresponding activity hotness threshold for one or more methods, and identifying one or more methods as meeting said activity hotness threshold for input to said controller device.*** (see Column 2, Lines 52-60, "In the Self system,

the determination of whether to re-compile previously compiled code is made based on how many times a specific portion of compiled code, such as a **method**, has been called. If the method has been invoked more times than a fixed limiting value, then the method is **re-compiled**. . The fixed limiting value is essentially a fixed **threshold**, which reflects the number of times the method is to be **invoked** before the method is re-compiled to increase efficiency in execution."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to compare method activations against a corresponding threshold. The modification would have been obvious because one of ordinary skill in the art would have been motivated to improve the efficiency of the system by recompiling the method that was invoked more than a fixed threshold times.

As Per Claim 6, the rejection of claim 3 is incorporated and further Krishnaswamy didn't disclose adaptively adjusting threshold.

However, Holzle teaches ***controller device adaptively adjusts said sampling size threshold.*** (see abstract, "This threshold value is **periodically adjusted** to keep the compilation and the interpretation overheads within acceptable ranges."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to adaptively adjust threshold. The modification would have been obvious because one of ordinary skill in the art would have been motivated to keep the compilation and the interpretation overheads within acceptable ranges.

As Per Claim 7, the rejection of claim 4 is incorporated and further Krishnaswamy didn't disclose adaptively adjusting threshold.

However, Holzle teaches ***mechanism for adapting said sampling size threshold in accordance with an amount of recompilation that occurs.*** (see abstract, "This threshold value is **periodically adjusted** to keep the compilation and the interpretation overheads within acceptable ranges.") and (see Column 2, Lines 52-60, "In the Self system, the determination of whether to re-compile previously compiled code is made based on how many times a specific portion of compiled code, such as a **method**, has been called. If the method has been invoked more times than a fixed limiting value, then the method is **re-compiled**. . The fixed limiting value is essentially a fixed **threshold**, which reflects the number of times the method is to be **invoked** before the method is re-compiled to increase efficiency in execution."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to adaptively adjust threshold in accordance within an amount of recompilation that occurs. The modification would have been obvious because one of ordinary skill in the art would have been motivated to keep the compilation and the interpretation overheads within acceptable ranges.

As Per Claim 8, the rejection of claim 2 is incorporated and further Krishnaswamy didn't disclose adaptively adjusting threshold.

However, Holzle teaches ***controller device dynamically adjusts said activity hotness threshold to adapt to a current behavior of the executing computer program.***

(see abstract, "This **threshold** value is periodically **adjusted** to keep the compilation and the interpretation overheads within acceptable ranges."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to adaptively adjust threshold. The modification would have been obvious because one of ordinary skill in the art would have been motivated to keep the compilation and the interpretation overheads within acceptable ranges.

As Per Claim 11, the rejection of claim 4 is incorporated and further Krishnaswamy didn't disclose minimizing future running time.

However, Holzle teaches ***an executing method, said controller device including a mechanism for identifying a recompilation level that minimizes expected future running time of a recompiled program.*** (see Column 2, Lines 39-47, "Second compiler 122 is often a slower compiler than first compiler 116, although code compiled using second compiler 122 typically executes more efficiently than code compiled using first compiler 116. Therefore, the determination of when to re-compile highly executed compiled code 120 generally involves a trade-off between additional compilation overhead, with respect to overall run-time, and the **improved efficiency** afforded by re-compiled highly executed code 124."; In other word, "Minimizing the running time" is inherent because the execution efficiency has been improved). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to minimize future running time. The modification would have been obvious because one of ordinary skill in the art would have been motivated to improve the efficiency of execution.

As Per Claim 18, the rejection of claim 1 is incorporated and further Krishnaswamy didn't disclose an interpreter device.

However, Holzle teaches ***execution environment includes an interpreter device.***

(see Column 5, Lines 66-67 to Column 6, Line 1, " When byte codes 144 are provided to computer system 146, byte codes 144 may be processed with an **interpreter** 148."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to have the execution environment to include an interpreter device. The modification would have been obvious because one of ordinary skill in the art would have been motivated to combine the optimization of Krishnaswamy with various type of system, including the system of Holzle to meet the needs of various types of systems.

In regard to ***Claim 21***, the rejection of claims 20 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 3.

In regard to ***Claim 22***, the rejection of claims 21 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 4.

In regard to ***Claim 23***, the rejection of claims 22 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 5.

In regard to ***Claim 24***, the rejection of claims 21 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 6.

In regard to ***Claim 25***, the rejection of claims 22 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 7.

In regard to ***Claim 26***, the rejection of claims 22 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 8.

As Per Claim 27, the rejection of claims 26 is incorporated and further Krishnaswamy didn't explicitly disclose determine amount of methods. However, HÖLZLE teaches ***determining an amount of methods characterized as meeting, said threshold criteria after one or more sampling periods;***(see Column 3, Lines 30-36, "An invocation tracker tracks the number of invocations of the selected method. When the number of invocations of the selected method exceeds a **threshold** value, the method is compiled. By independently tracking the usage of various methods or other code segments, a more **intelligent decision** can be made as to **which methods** should be compiled, and which methods should remain interpreted."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to determine amount of methods. The modification would have been obvious because one of ordinary skill in the art would have been motivated to verify the degree of system optimization.

Krishnaswamy further didn't explicitly disclose comparing said amount to a limit. However, HÖLZLE teaches ***comparing said amount to a limit:*** (see Column 7, Lines 16-21, "The process 202 begins at step 204 where a threshold monitor is started. The threshold monitor executes concurrently with a computer program, as will be described below, and keeps **track** of a threshold which indicates the number of times a particular method is to be executed before the method may be considered for compilation."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to compare said amount to a limit. The modification would have been obvious because one of ordinary skill in the art would have been motivated to decide whether data collected exceeds the threshold limit. ***and,***

Krishnaswamy further didn't explicitly disclose changing threshold value. However, HÖLZLE teaches ***in response to said comparing, one of: decreasing said activity hotness threshold if said limit is not met, and increasing the threshold if said limit is met or exceeded in said one or more sampling periods.*** (see Column 13, Lines 6-24, "If the interpretation overhead is greater than the maximum desired interpretation overhead, **then the threshold is decreased in step 712**, using any suitable process such as the process described below with respect to FIG. 8. Decreasing the threshold generally enables more methods to be compiled, thereby reducing the number of methods which are interpreted. Alternatively, if the interpretation overhead is determined to be less than the maximum desired interpretation overhead, then in step 718, it is determined whether the interpretation overhead falls below the minimum desired interpretation overhead. When the interpretation overhead is less than the minimum desired interpretation overhead, then **the threshold is increased in step 716**. In the event that the

interpretation overhead is within the range of acceptable interpretation overheads defined by the minimum desired interpretation overhead and the maximum desired interpretation overhead, then the process of executing the threshold monitor continues in step 702 where a new timer signal is received from the operating system."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Holzle into the system of Krishnaswamy to change threshold value. The modification would have been obvious because one of ordinary skill in the art would have been motivated to tune-up the system optimization.

In regard to ***Claim 30***, the rejection of claims 22 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 11.

In regard to ***Claim 36***, the rejection of claims 19 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 18.

Claim 37 is the computer-readable medium claim corresponding to the method claim 19 and is rejected under the same reason set forth in connection of the rejection of claim 19. Further Krishnaswamy disclosed ***computer-readable medium*** (see Column 4, Lines 35-40, "The I/O controller 80 controls access to and information from external devices such as keyboards 140, monitors 150, **permanent storage 160**, and a removable media unit 170. In addition, the computer system may be connected through a network connection 180 to other computer systems.").

In regard to ***Claim 38***, the rejection of claims 37 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 20.

In regard to ***Claim 39***, the rejection of claims 37 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 21.

In regard to ***Claim 40***, the rejection of claims 39 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 22.

In regard to ***Claim 41***, the rejection of claims 40 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 23.

In regard to ***Claim 42***, the rejection of claims 39 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 24.

In regard to ***Claim 43***, the rejection of claims 40 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 25.

In regard to ***Claim 44***, the rejection of claims 40 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 26.

In regard to ***Claim 45***, the rejection of claims 44 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 27.

In regard to ***Claim 48***, the rejection of claims 40 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 30.

In regard to ***Claim 54***, the rejection of claims 37 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 36.

4. Claims 9, 28 and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy in view of Holzle al. (US Patent No. 5,995,754) hereafter Holzle further in view of O'Donnell (US Patent No. 6,374,369).

As Per Claim 9, the rejection of claim 4 is incorporated and further Krishnaswamy and Holzle didn't disclose to insert intrusive profiling. However, O'Donnell teaches ***controller instructs the recompilation subsystem to insert intrusive profiling for one or more identified methods.*** (see Column 1, Lines 13-22, "Software developers, especially those developing time and/or performance critical software, such as real-time programs for embedded processors or driver software for peripherals, need to determine where and how much time is spent in the various parts (routines) of their software in order to determine how to further optimize their software to meet necessary performance and response requirements. Such performance analysis, also known by the term **"profiling," has traditionally been one that is intrusive** to the code being analyzed."). Therefore, it would have been obvious to one of ordinary skill in the art at the

time the invention was made to incorporate the teaching of O'Donnell into the system of Holzle and Krishnaswamy to insert intrusive profiling. The modification would have been obvious because one of ordinary skill in the art would have been motivated to determine how to optimize their software.

In regard to ***Claim 28***, the rejection of claims 22 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 9.

In regard to ***Claim 46***, the rejection of claims 40 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 28.

5. Claims 10, 29 and 47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy in view of Holzle al. (US Patent No. 5,995,754) hereafter Holzle further in view of Benitez et al. (US Patent No. 6,189,141) hereafter Benitez.

As Per Claim 10, the rejection of claim 4 is incorporated and further Krishnaswamy disclosed

-an optimization level indicating a degree of optimization. (see Column 5, Lines 43-51, "The **dynamic optimization helper 260** in kernel memory space operates similarly to the **dynamic optimization helper 230** in shared user memory space. It collects profile data regarding the executing computer operating system kernel 220, processes that profile data, and

provides it to the code-rewriting kernel module 240. The kernel module 240 analyzes the profile data and **generates optimized translations** of portions of the computer operating system kernel 220 from the profile data.") In other word, since the optimization is performed dynamically using the profile data, various levels of optimization is inherently performed. They depend on the result of the analysis of the profile data.

Krishnaswamy and Holzle didn't disclose a method identifier. However, Benitez teaches ***an identifier of a method to be optimized;*** (see Column 1, Lines 13-22, "The invention includes a **hot trace identifier** to identify frequently executed traces of instructions and a hot trace instrumenter to instrument such frequently executed traces so that control flow through them may be recorded. If the amount or rate of control flow through a frequently executed trace exceeds a threshold value, a hot trace selector is invoked to select a hot trace of original instructions including those of the frequently executed trace. The **hot trace** may be **dynamically optimized**."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Benitez into the system of Holzle and Krishnaswamy to have a method identifier. The modification would have been obvious because one of ordinary skill in the art would have been motivated to easily differentiate the "hot" methods from other type of methods to facilitate the optimization process.

In regard to ***Claim 29***, the rejection of claims 24 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 10.

In regard to ***Claim 47***, the rejection of claims 42 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 10.

6. Claims 12, 31 and 49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy in view of Holzle al. (US Patent No. 5,995,754) hereafter Holzle, further in view of Ronstrom (US Pub, No: US 2002/0010913 A1).

As Per Claim 12, the rejection of claim 11 is incorporated and further Krishnaswamy and Holzle didn't disclose expected time and cost relations with/without method recompiled. However, Ronstrom teaches ***mechanism for determining an expected time Ti the program will spend executing a method "m" if said method is not recompiled; mechanism for determining a cost Cj of recompiling said method at an optimization level `j", for i < j < N; and, mechanism for determining an expected time T. the program will spend executing said method in the future, if said method is rccompiled at level "j"; and, comparison mechanism for evaluating the expression Cj + Tj < Ti, whereby said controller device decides to one of: generate compilation plan for directing recompilation of "m" at level "j" if said expression is true, and, not recompile if said expression is false.*** (see Section [0021], "Therefore, one idea underlying the present invention is to use the jump memory to provide profiling information as well as information useful for debugging and testing purposes. The **costs** of continuously writing to the jump memory are justified by the use of the information stored therein for the support of the subsequent compiling process and for debugging purposes."), and (see Section [0027], "According to a preferred embodiment, the recompilation step is carried out a **plurality of times**

to achieve an iterative improvement of the target program.") and (see section [0029], "According to yet another preferred embodiment of the present invention, it is checked whether a recompilation should be carried out each time an analysis interval for the analysis of the generated **execution statistics** has elapsed. Here, this analysis interval may be specified either as **maximum number of steps of the target program** to be executed or as time interval."), and (see Section [0066], "In a further step S16 it is determined whether the overall execution of the input program should be terminated or not. If this is not the case there follows a step S18 to carry out an analysis of the generated execution statistics. This analysis S18 of the execution statistics forms the basis for the determination of step S20, i.e. whether a further **optimisation** is necessary or not. If it is determined in step S20 that a further optimisation is necessary step S22 will be carried out to recompile the currently existing target program while using the execution statistics as compiler support. "), and (see Section [0067], "As shown in FIG. 2, the recompilation step S22 is carried out a **plurality of times** to achieve an iterative improvement of the target program. Also, a recompilation step S22 is carried out each time an **analysis** interval for the analysis of the generated execution statistics has elapsed."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Ronstrom into the system of Holzle and Krishnaswamy to have the expected time and cost relations with/without method recompiled. The modification would have been obvious because one of ordinary skill in the art would have been motivated to fine tune the system optimization by recompilation if the cost is lower.

In regard to ***Claim 31***, the rejection of claims 30 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 12.

In regard to ***Claim 49***, the rejection of claims 48 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 12.

7. Claim 13 is rejected under 35 U.S.C. 103(a) as being unpatentable over Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy in view of Alpern et al. "The Jalapeño Virtual Machine", IBM System Journal, Vol 39, No 1, February 2000, hereafter Alpern.

As Per Claim 13, the rejection of claim 1 is incorporated and further Krishnaswamy didn't disclose method prologue and back edge yield points. However, Alpern teaches ***raw profile data samples are taken at method prologue and back edge yield points.*** (see Page 222, left hand column, Line2 35-46, "The code produced by all three compilers must satisfyJalapen˜o's calling and preemption conventions. They ensure that threads executing the methods they compile will respond in a timely manner to attempts to preempt them. Currently, explicit yield points are compiled into **method prologues**. Eventually, **yield points** will be needed on the "**back edges**" of loops that cannot be shown to contain other yield points."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Alpern into the system of Krishnaswamy to use method prologue and back edge yield points. The modification would have been obvious because one of ordinary skill in the art would have been motivated to take profile data samples.???

8. Claims 14-17, 32-35 and 50-53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Krishnaswamy et al. (US Patent No. 6,622,300) hereafter Krishnaswamy in view of HÖLZLE et al, : "Reconciling Responsiveness with Performance in Pure Object-Oriented Languages", ACM Transactions on Programming Languages and Systems, Vol. 18, No. 4, July 1996, pp. 355-400, hereafter HÖLZLE.

As Per Claim 14, the rejection of claim 1 is incorporated and further Krishnaswamy didn't disclose online feedback, directed optimizations. However, HÖLZLE teaches ***controller instructs the recompilation subsystem to perform online feedback, directed optimizations based on feedback from the current executing program*** (see Page 362, Table I, Systems Used for Benchmarking, "SELF-93 The current SELF system using dynamic recompilation and type feedback;methods are compiled by a fast non-optimizing compiler first, then recompiledwith the optimizing compiler if necessary."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of HÖLZLE into the system of Krishnaswamy for feedback-directed optimizations. The modification would have been obvious because one of ordinary skill in the art would have been motivated to increase system performance.

As Per Claim 15, the rejection of claim 14 is incorporated and further Krishnaswamy didn't disclose call context information. However, HÖLZLE teaches ***raw profile data samples***

relate to call context information associated with methods called by said program, said feedback comprising said call context information. (see Page 372, Lines 5-8, "Then, the application is run with one or more test inputs that are representative of the expected inputs for production use. Finally, **the collected type and profiling information** is **fed back** to the compiler to produce the final optimized code."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of HÖLZLE into the system of Krishnaswamy for call context information. The modification would have been obvious because one of ordinary skill in the art would have been motivated to use feedback information to optimize system.

As Per Claim 16, the rejection of claim 14 is incorporated and further Krishnaswamy didn't disclose program variable value. However, HÖLZLE teaches ***raw profile data samples relate to current program variable values, said feedback comprising a subset of values assigned to said variables during program execution.*** (see Page 375, Lines 20-27, "SELF-93 approximates the ideal policy by using ***invocation counts*** to drive recompilation. Assuming that past behavior predicts future behavior, methods are recompiled if their invocation counter exceeds a certain limit. Each unoptimized method has **its own counter** that is incremented in the method prologue. When the counter exceeds the limit, the recompilation driver is invoked to decide which method (if any) should be recompiled. If the method overflowing its counter isn't recompiled, its counter is reset to zero."). The Examiner interprets "its own counter" is a "variable". Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of HÖLZLE into the system of

Krishnaswamy for update counter. The modification would have been obvious because one of ordinary skill in the art would have been motivated to decide which method (if any) should be recompiled.

As Per Claim 17, the rejection of claim 14 is incorporated and further Krishnaswamy didn't disclose program variable value. However, HÖLZLE teaches raw ***profile data samples relate to control flow execution within a method, said feedback comprising execution frequency of control flow paths within said executing method.*** (see Page 375, Lines 20-27, "SELF-93 approximates the ideal policy by using *invocation counts* to drive recompilation. Assuming that past behavior predicts future behavior, methods are recompiled if their invocation **counter exceeds a certain limit**. Each unoptimized method has its own counter that is incremented in the method prologue. When the counter exceeds the limit, the recompilation driver is invoked to decide which method (if any) should be recompiled. If the method overflowing its counter isn't recompiled, its counter is reset to zero."). The Examiner interprets "a certain limit" is a "frequency". Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of HÖLZLE into the system of Krishnaswamy to update counter. The modification would have been obvious because one of ordinary skill in the art would have been motivated to decide which method (if any) should be recompiled.

In regard to ***Claim 32***, the rejection of claims 19 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 14.

In regard to ***Claim 33***, the rejection of claims 32 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 15.

In regard to ***Claim 34***, the rejection of claims 32 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 16.

In regard to ***Claim 35***, the rejection of claims 32 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 17.

In regard to ***Claim 50***, the rejection of claims 37 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 32.

In regard to ***Claim 51***, the rejection of claims 50 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 33.

In regard to ***Claim 52***, the rejection of claims 50 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 34.

In regard to ***Claim 53***, the rejection of claims 50 is incorporated and is rejected under the same reason set forth in connection of the rejection of claim 35.

Response to Arguments

9. Applicant's arguments with respect to claims 1-54 have been considered but they are not persuasive.

In the remarks, the applicant argues that:

1) As for claim 1, Applicant argues the present invention is not limited to kernel space only. (E.g. see page 18, line 18).

Examiner's response:

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., the present invention is not limited to kernel space only. (E.g. see page 18, line 18)) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

2) As for claims 1, 10, 29 and 47, Applicant argues that Krishnaswamy does not cover "level of optimization" (E.g. see page 19, lines 5-10 and page 22, lines 13-14).

Examiner's response:

Examiner disagrees with applicant's assertion that Krishnaswamy does not cover "level of optimization". In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., "characterizations thereof (See page 21, line 27 – page 22 line 15, of the present patent

application)" (E.g. see page 19, lines 8-9)) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). Furthermore, after optimization, the outcome is at a different level/degree than before optimization is applied.

3) As for claim 1, Applicant argues the third element set forth in Claim 1, describes the process of implementing the compiler to actually perform a level of program optimization (i.e., a compilation).

Examiner's response:

Examiner disagrees with applicant's assertion that Krishnaswamy doesn't teach describes the process of implementing the compiler to actually perform a level of program optimization. In previous Office Action mailed on 10/28/2003, it shows that (see Column 5, Lines 51-58, "The code-rewriting kernel module 240 then writes the optimized translations into a code cache 270 in the kernel memory space. The kernel module 240 also inserts jump instructions into the computer operating system kernel 220 to switch execution flow of control to the optimized translations whenever an optimized instruction within the computer operating system kernel 220 is called."). (Emphasis added). The "writes ... code caches 270" means the output of optimization is produced, that is a compilation must have been performed otherwise it will not work; and the "whenever an optimized instruction ... called" means that the optimization is done and running.

4) As for Claim 19, Krishnaswamy does not teach a sampling technique as implemented in the present invention (E.g. see page 19, last line).

Examiner's response:

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., Krishnaswamy does not teach a sampling technique as implemented in the present invention (E.g. see page 19, last line)) are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

5) Holzle does not teach proper counter in claim3, 1st limitation :"a mechanism for counting a number of sample that are taken from the executing program" (E.g. see page 20, last paragraph).

Examiner's response:

Examiner disagrees that Holzle does not teach "a mechanism for counting a number of samples that are taken from the executing program". Applicant acknowledge that in the Holzle system, a counter is incremented unconditionally every time particular points (number of samples) in the program execution are reached. (E.g. see page 20, last paragraph). In the palin claim language, there is no limition whether the counter is incremented conditionally or unconditionally.

6) As for claims 9, 28 and 46, the teaching of O'Donnell could not be used to automatically perform a corresponding method step of inserting intrusive profiling. (E.g. see page 22, lines 2-5).

Examiner's response:

Examiner disagrees with applicant's assertion that the teaching of O'Donnell could not be used to automatically perform a corresponding method step of inserting intrusive profiling. O'Donnell teaches a system for implementing the profiling method in both software and hardware (E.g. see Abstract) and optimizing compiler (E.g. see col. 13:36-40). There is no mention in the claim that "inserting instrusive profiling" must be "automatically perform (no human intervention)".

7) As for claim 16, the variables in Holzle2 are not created by the programmer. (E.g. see page 22, lines 21-22).

Examiner's response:

Examiner disagrees with applicant's assertion that the variables in Holzle2 are not created by the programmer. In fact, in the previous Office Action mailed on 10/28/2003, it states that "Each unoptimized method has its own counter that is incremented in the method prologue. (E.g. see page 375, lines 20-27)" (Emphasis added). The counter is a variable and it is related to that method/programmer.

8) As for claim 17, the invocation counters do not correspond to control flow paths within an executed method, which represents semantics at the application level. (E.g. see page 23, lines 6-8).

Examiner's response:

Examiner disagrees with applicant's assertion that the invocation counters do not correspond to control flow paths within an executed method, which represents semantics at the application level. In fact, in the previous Office Action mailed on 10/28/2003, it states that "When the counter exceeds the limit, the recompilation driver is invoked to decide which method (if any) should be recompiled. (E.g. see page 375, lines 20-27)" (Emphasis added). The "limit" is a "level" and the "decide .. recompiled" is a "flow of control path".

Conclusion

10. **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Correspondence Information

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kuo-Liang J Tang whose telephone number is 703-305-4866. The examiner can normally be reached on M-F 8:30 to 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q Dam can be reached on 703-305-4552.

Any response to this action should be mailed to:

Commissioner of Patents and Trademarks

Washington, D.C. 20231

or faxed to:

(703) 872-9306.

Kuo-Liang J. Tang

Software Engineer Patent Examiner

ANTONY NGUYEN-BA
PRIMARY EXAMINER